# Mikail

**DataBaseUpdateManagerModel**

+CurrentCity()

+SetTimerInitial()

+SetTimerContinious()

+IsConnectionAvailable()

+UpdateWeather()

give info to update current weather data

**DataBaseManagerModel**

+ WeatherToDisplayListModel WeatherToDisplays

+ string CityName

- WeatherInfoListModel weatherInfos

- string CountryZip

- int CityId

- string MS

- string CELSIUS

- string HUM

- string FilePath

+DataBaseManagerModel()

+GetDataFromOpenWeather(string CityName)

```
-GetWeather(string CityName)
+SaveIntoDatabase()
+LoadFromDatabase(string CityName)
+GetCitiesByLetters(string LettersForCityname)
-InsertIntoDataBase(WeatherInfoModel weatherInfo)
-CheckIfCurrentDataExist(WeatherInfoModel weatherInfo)
-UpdateDataBase(WeatherInfoModel weatherInfo)
```

gets data from internet

inherits property changed    property observablecollection contains e.g. temp., humidity,...

**APIManagerModel**

-string API_KEY

-WeatherInfoListModel GetWeatherInfos(string JSONContent)
-ToWeatherInfoModel WeatherInfoModel(JSONListItem item)
-string WindDirConverter(double winddir)
+Task GetWeather(string location)

stores into database

**INotifyPropertyChanged**

**WeatherToDisplayListModel**

helper class to deserialize data from JSON

convert received JSON into WeatherInfoListModel

inherits observablecollection of WeatherToDisplayModel

**JSONModel**

+class JSONResponce
+class JSONListItem
+class JSONMainInfo
+class JSONWeatherType
+class JSONWeatherWindInfo
+class JSONCity

**WeatherInfoListModel**

**WeatherToDisplayModel**

+ string WeatherDescription
+ string Weekday
+ string WeatherIcon
+ string MaxTemperature
+ string MinTemperature
+ string AvgTemperature
+ string WindSpeed
+ string Winddirection
+ string Humidity
+ BitmapImage WeatherImageSource

+WeatherToDisplayModel(string weatherdesc, ...)

**JSONResponce**

~List<JSONListItems> items
~JSONCity City

**JSONListItem**

+List<JSONWeatherType> WeatherTypes
+JSONWeatherWindInfo WeatherWindInfo
+string DateTime

**JSONMainInfo**

+double Temp_min
+double Temp_max
+double Humidity

**JSONWeatherType**

+string Description
+string Icon

**JSONWindInfo**

+double WindSpeed
+double WindDirection

**JSONCity**

+string Name
+string CountryZip

inherits List of WeatherInfoModel

**WeatherInfoModel**

+string WeatherDescription
+string WeatherIcon
+DateTime WeatherDayTime
+double MaxTemperature
+double MinTemperature
+double WindDirection
+string WindDirectionAsString
+double WindSpeed
+double Humidity

+WeatherInfoModel(string weatherdesc, ...)